Predicting Premium Product Clicks in Online Shopping

Insights from UCI Online Shopping Clickstream Dataset

Barbara Kotlan
Lally School of Management
Rensselaer Polytechnic Institute
Troy, NY, USA
kotlab@rpi.edu

ABSTRACT

This report focuses on using machine learning classification to predict whether a selected product is above its own category average price. A product above this average can be referred to as a premium product. Predicting premium products will display which factors impact premium clicks. These factors can inform marketing strategies and strategic decisions regarding premium products, helping businesses optimize product placement and product sales. The chosen dataset is from the UCI repository, containing Clickstream Data for Online Shopping. The data is from April 2008 to August 2008. Each row in the dataset corresponds to a single click on the online site. The data was preprocessed with binarization, one-hot encoding, and dropping NaN values. The models used to predict premium products were Random Forest Classifier, Gradient Boosting Classifier, and CatBoost Classifier. Each model was tested with multiple test sizes, ranging from 0.2 to 0.4 in the train, test, and split. The CatBoost model achieved the best performance with a test size of 0.3, being better than all other models. It had an accuracy score of 0.932, a precision of 0.933, a recall of 0.928, a F1-score of 0.931, a roc-auc of 0.989, and a pr-auc of 0.989. Key predictors that impacted premium products were the color of product, location on webpage, site page number, product category, and model in photography. These findings highlight the factors that influence premium product selection, enabling the optimization of customer targeted marketing strategies.

KEYWORDS

clickstream analysis, premium products, classification models, online shopping behavior, CatBoost, Gradient Boosting, Random Forest, customer browsing behavior, purchase prediction

EXECUTIVE SUMMARY

1. Data Description

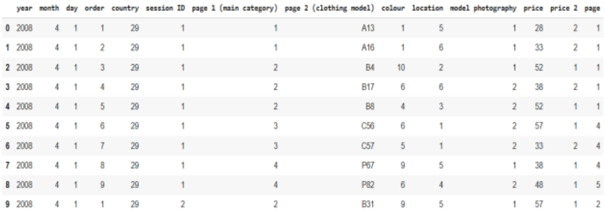
This dataset is called Clickstream Data for Online Shopping from the UCI repository [1]. It stores customer browsing behavior data from an online clothing store. Each row represents a singular click on the online set. The data covers five months of online activity on the site in 2008. The objective is to analyze the data to identify behavioral patterns, predict purchases, and understand factors that impact site clicks.

The data has 165,474 instances and 14 features. The features include:

- year - the year of the data recorded (2008)
- month - the month of the data recorded (April-August)
- day - the day of the data recorded
- order - the sequence of the click in its session
- Country - the country of origin of the IP address, stored as a mapped number
- session ID – an identification of the site session
- page 1 (main category) - the clothing category of the product, stored as a mapped number
- page 2 (clothing model) - the unique product code
- color - the color of the selected product, stored as a mapped number
- location - the location of the product photo on the screen, stored as a mapped number
- model photography - whether the product contains a model, stored as (1, 2)
- price - the price of the selected product in USD
- price 2 – whether the product price is higher than the category average, stored as (1, 2)
- page - the number of the page within the e-store site

The class label would be price 2, a variable informing whether the price of a particular product is higher than the average price for the entire product category. This is used to analyze what clicks are on premium products. Knowing the color, location, and other attributes of clicked products can aid the company in knowing how to sell and market premium products to customers.

Table 1. Raw clickstream data sample



	year	month	day	order	country	session ID	page 1 (main category)	page 2 (clothing model)	colour	location	model photography	price	price 2	page
0	2008	4	1	1	29	1	1	A13	1	5	1	28	2	1
1	2008	4	1	2	29	1	1	A16	1	6	1	33	2	1
2	2008	4	1	3	29	1	2	B4	10	2	1	52	1	1
3	2008	4	1	4	29	1	2	B17	6	6	2	38	2	1
4	2008	4	1	5	29	1	2	B8	4	3	2	52	1	1
5	2008	4	1	6	29	1	3	C56	6	1	2	57	1	4
6	2008	4	1	7	29	1	3	C57	5	1	2	33	2	4
7	2008	4	1	8	29	1	4	P67	9	5	1	38	1	4
8	2008	4	1	9	29	1	4	P82	6	4	2	48	1	5
9	2008	4	1	1	29	2	2	B31	9	5	1	57	1	2

2. Data Observations

The initial observations include that the dataset contains 165,474 clicks with 14 total columns. Most features are numeric, while country, page 1 (main category), page 2 (clothing model), colour, and location are categorical values mapped to numbers. There are no missing values in the dataset.

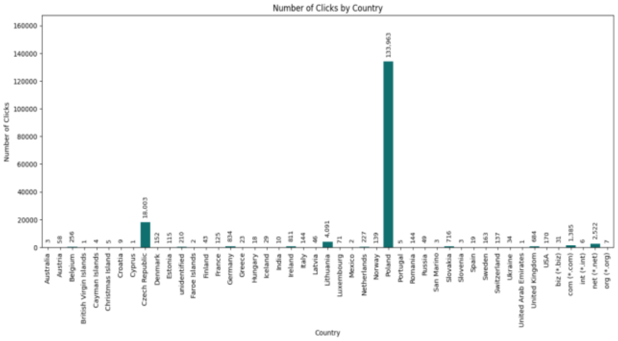


Figure 1. Number of clicks by country. Countries are ordered by their numeric code (1-47).

As shown in **Figure 1**, the dataset contains clicks from 47 countries and/or domains. The distribution clearly displays how Poland has the highest number of click by far, with the Czech Republic trailing far behind as the next highest. The remaining countries account for significantly fewer clicks. There is also a small subset that is not attached to a country but rather has a domain.

The same type of mapping categorical variables occurs similarly for features page 1 (main category), page 2 (model photography), colour, and location. This means each number in these columns represents a string, rather than just the displayed value.

3. Data Preprocessing and Modeling

The data was preprocessed with binarization, one-hot encoding, dropping NaN values, and feature engineering two variables. The models used to predict premium products were Random Forest Classifier, Gradient Boosting Classifier, and CatBoost Classifier. Each model was tested with multiple test sizes, ranging from 0.2 to 0.4 in the train, test, and split. The CatBoost model achieved the best performance with a test size of 0.3, being better than all other models. It had an accuracy score of 0.932, a precision of 0.933, a recall of 0.928, a F1-score of 0.931, a roc-auc of 0.989, and a pr-auc of 0.989. Key predictors that impacted premium products were the color of product, location on webpage, site page number, product category, and model in photography.

BENCHMARKING OTHER SOLUTIONS

This is a dataset from the UCI repository; therefore, there are no available solutions completed by others, and this section is skipped.

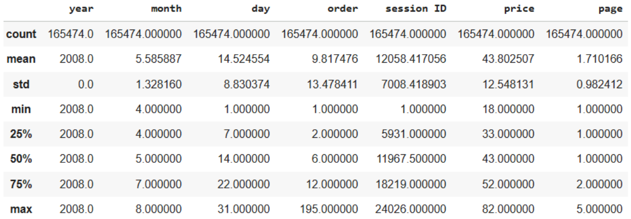
DATA DESCRIPTION AND INITAL PROCESSING

1. Data Statistics

This section focuses on the initial preprocessing and statistical characterization of these features, to prepare the dataset for various modeling.

The dataset contains fourteen features, half of which are numerical. The numerical features include year, month, day, order, session ID, price, and page. The remaining seven are all categorical, six of which are mapped to numerical values. This includes country (mapped from 1-47), page 1 which represents the product's main category (mapped from 1-4), page 2 which represents the clothing model or product ID, colour (mapped from 1-14), location on page (mapped from 1-6), model photography (mapped from 1-2), and price 2 (mapped from 1 to 2).

Table 2. Summary statistics of numerical attributes



	year	month	day	order	session ID	price	page
count	165474.0	165474.000000	165474.000000	165474.000000	165474.000000	165474.000000	165474.000000
mean	2008.0	5.585887	14.524554	9.817476	12058.417056	43.802507	1.710166
std	0.0	1.328160	8.830374	13.478411	7008.418903	12.548131	0.982412
min	2008.0	4.000000	1.000000	1.000000	1.000000	18.000000	1.000000
25%	2008.0	4.000000	7.000000	2.000000	5931.000000	33.000000	1.000000
50%	2008.0	5.000000	14.000000	6.000000	11967.500000	43.000000	1.000000
75%	2008.0	7.000000	22.000000	12.000000	18219.000000	52.000000	2.000000
max	2008.0	8.000000	31.000000	195.000000	24026.000000	82.000000	5.000000

A summary of the numerical features is displayed in **Table 2**. This contains the truly numerical features of the dataset. It excludes categorical features which have been mapped to numerical codes. For the numerical features, the summary statistics display the distribution of each feature. For year, there is only one value being 2008. For month, there are multiple values between 4 and 8 representing the months. The mean is 5.59, displaying a somewhat even distribution of months. The day has values ranging from 1 to 31. The average is 14.52, which is expected for evenly distributed days. Order, which is the number click in its session, has a minimum of 1 click and maximum of 195 clicks. This is a very big range, but the mean of 9.82 shows how it is right-skewed. The session ID ranges from 1 to 204026, representing all of the sessions that occurred in the dataset. The average and quartiles are not useful, as the numbers act as an ID and not a value. Product price ranges from 18 to 82, with an average of 43.80. Lastly, page of the site ranges from 1 to 5, with the mean being 1.71. The mean and the quartiles display how the majority of clicks happen on earlier pages.

2. Data Characterizations

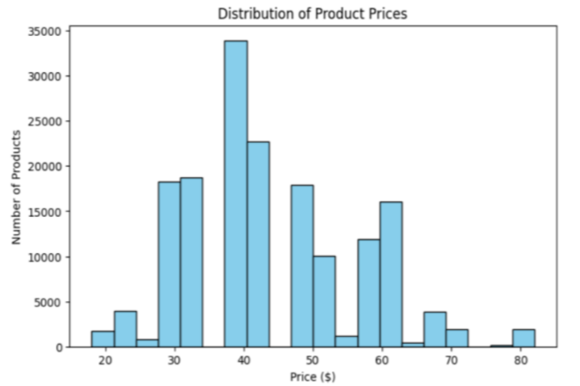


Figure 2. The distribution of product prices

The price of products is vital in this report, as we aim to predict premium, high-priced, products. **Figure 2** displays the distribution of product prices from clicks. Each price point corresponds to the number of clicks where an item was at that price. The distribution is slightly right-skewed, with slightly more clicks being on mid-to-lower priced items. Therefore, the dataset has slightly more non-premium products than premium products. To fully see the extent of the premium product distribution we are predicting, **Figure 3** is a bar graph with the percentage of both premium and non-premium products based on the prediction variable. Although there is a difference, 51.2% compared to 48.8% shows that there is a very even split. The distribution is favorable for predictive modeling due to minimal class imbalance. Such balance helps prevent bias towards either category or ensure a more reliable training model.

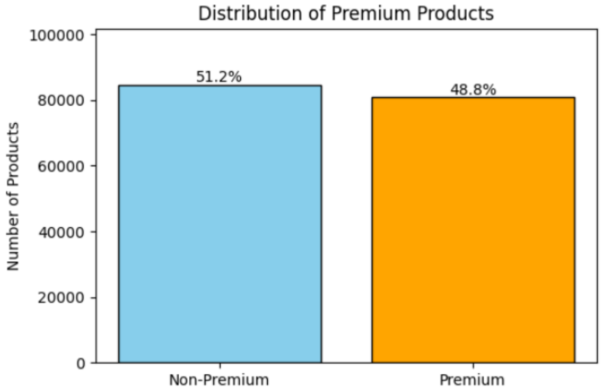


Figure 3. Distribution of Premium Products. Based on variable Price 2, displays percentage of premium products.

As for the category of products (known as page 1), there are four labeled values. Numerical value 1 is trousers, 2 is skirts, 3 is blouses, and 4 is products on sale. The distribution is shown in **Figure 4**, with a generally even split between each category. Trousers has more datapoints than the rest of the categories, however the amount is not significant.

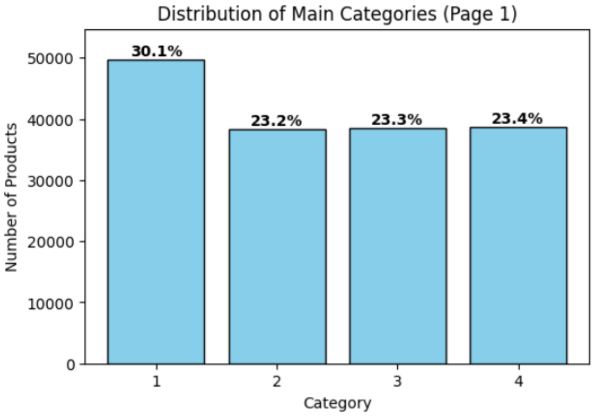


Figure 3. The distribution of page 1 (main categories)

The categories can be useful when looking and products, as well as the actual product ID distribution. This distribution will display how many clicks are on each product. There are 217 total products with click data. **Figure 4** contains the top 10 products that have the most data. The top product has over 3,500 clicks, which is still 0.0% of all clicks. This is because there are more than 100,000 clicks in the dataset. Although these are the top clicked on products, they are a small fraction of overall user activity. It displays a highly diverse distribution of engagement across all the various products. While this may not seem useful, it shows how the data is symmetrical and not skewed towards any certain products.

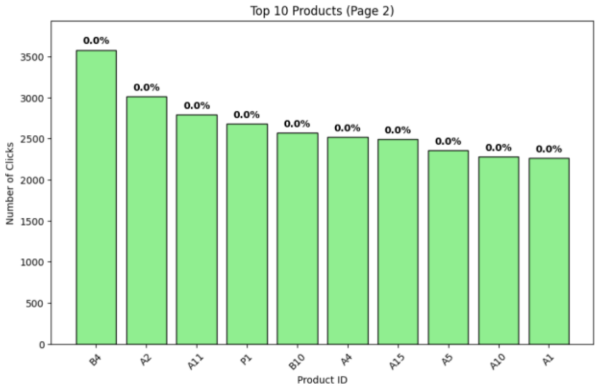


Figure 4. The top 10 clicked of product IDs

Although the product IDs are even, the color of products is not. There are 14 total recorded colors, each listed below corresponding to their number in **Table 2**.

Table 2. Color id to product color mappings

Number	Color
1	beige
2	black
3	blue
4	brown
5	burgundy
6	gray
7	green
8	navy blue
9	of many colors
10	olive
11	pink
12	red
13	violet
14	white

Using both **Table 2** and **Figure 4**, there is a visible gap between many colors. Black is the most prominent color, taking up 18% of all product clicks. Blue is a close second with 17.7% of clicks. Yet, there are colors with much less clicks. Burgundy is only 1.0% of all clicks, and navy blue is 1.6%. It is unclear whether there are less products of certain colors like navy blue and burgundy, or whether it is just the number of clicks that varies. There is a gap of knowledge about the color distribution of the dataset, which makes it hard to maximize the results related to color.

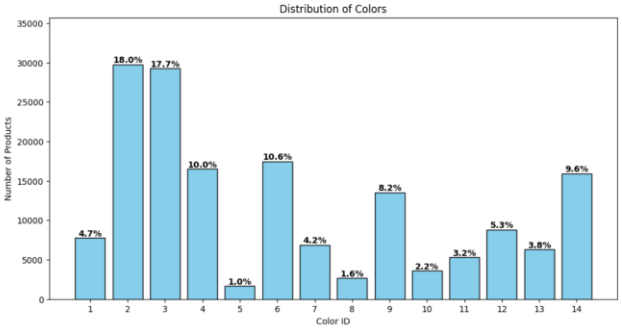


Figure 4. The distribution of colors

The location of products on the page is a feature that can be looked at for predicting premium products. As the baseline, **Figure 5** displays a roughly even distribution of page location. Each number corresponds to a location. Value 1 is top left of the page, 2 is the top middle, 3 is top right, 4 is bottom left, 5 is bottom middle, and 6 is bottom right. There are slightly less product clicks in the top and bottom right section of the page, with only 13.1% and 12.5% respectively. This is even enough to be a useful factor for prediction.

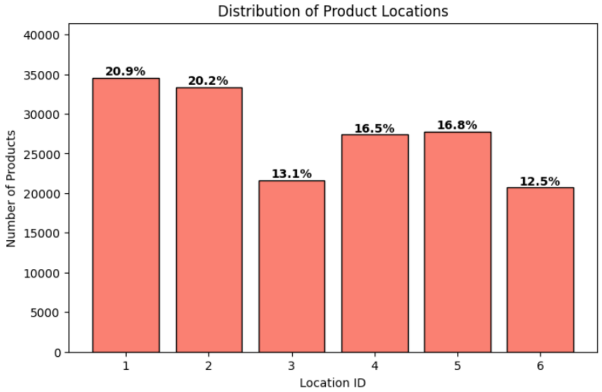


Figure 5. The distribution of product locations on the site page

The last feature to cover is model photography. There are two values – en face and profile. En face corresponds to a model facing towards the camera, while profile is a model photographed from a side profile. **Figure 6** shows how there are a large number of products with front-facing models.

This uneven distribution shows overall preference in product presentation, which influences engagement. The gap should be considered when analyzing the dataset to avoid bias is predictive tools.

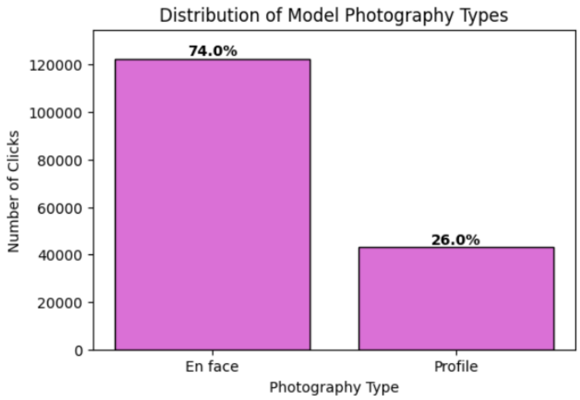


Figure 6. The distribution of model photography types

3. Data Preprocessing

Given all of these various features, it is important to properly prepare the dataset for any analysis. The preprocessing technique first involved checking for empty values. There were no missing values, so there was no handling needed. The next step was binarization of variables model photography and price 2, converting their values from 1 and 2 to 0 and 1. Afterwards, feature engineering was done to make use of more features in the dataset. Session length, the number of clicks in a session, was added to each row. As for the timestamp variables, they were used to create a day of the week feature. Finally, one-hot encoding was used to turn the categorical variables ('country', 'page 1 (main category)', 'page 2 (clothing model)', 'colour', 'location') into numerical variables. The data was then split into two versions, one with hot-encoding and one set without one-hot encoding to fit with the specific machine learning models.

MODELING

1. Model Descriptions

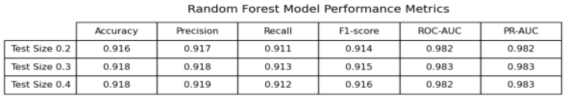
This data was analyzed using three machine learning models to classify the data. The first chosen model was a Random Forest classifier. Random Forest is an ensemble model which builds many decision trees to make predictions [2]. It is able to handle complex feature interactions that are non-linear. With averaging various decision trees, it is less sensitive to outliers. Randoms Forest provides feature importance, which is useful for the clickstream dataset. There is also no required normalization or scaling data when using this model. Gradient Boosting classifier was the next model used on the dataset. Gradient Boosting is also an ensemble model which creates sequential trees that learn from previous errors [3]. Since it learns from error, it is often a high performing model when tunned correctly. It is able to capture non-linear patterns, and it still has a feature importance category. The last model used to classify was CatBoost. CatBoost is an open-source model based on gradient boosting [4]. It has advantages of handing categorical variables and built-in overfitting prevention. Due to the number of categorial numerically mapped variables, CatBoost kept the dimensionality of the dataset lower.

2. Model Performance Metrics

To apply each model, the data was trained, tested, and split in various ways. Multiple test sizes were used to observe various results. The data was used during test sizes of 0.2, 0.3, and 0.4 for each model. Performance metrics of accuracy, precision, recall, F1-score, ROC-AUC, and PR-AUC were used to evaluate each model for each test size.

CatBoost outperformed both Random Forest and Gradient Boosting across nearly all metrics. As viewed in **Table 5**, CatBoost had an accuracy score of 0.933 for test size 0.3 and around the same for all other test sizes. Random Forest was a close second with an accuracy of 0.918 in **Table 3**. Gradient Boosting fell behind with an accuracy of 0.797 as displayed in **Table 4**. For recall, Gradient Boosting is significantly lower than all other models, with its highest score of 0.721. Random Forest had a recall of 0.913 as its highest, while CatBoost dominated with 0.928 as the highest. The same pattern appears with precision, F1-score, and ROC-AUC. Yet, for PR-AUC, Random Forest and Gradient Boosting both have a highest score of 0.983. This is also very close to the CatBoost score of 0.989. **Table 3, 4, and 5** all displayed a clear patten of CatBoost outperforming both Random Forest and Gradient Boosting.

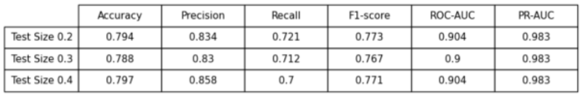
Table 3. Random Forest Model Performance Metrics



Random Forest Model Performance Metrics

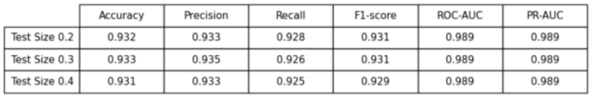
	Accuracy	Precision	Recall	F1-score	ROC-AUC	PR-AUC
Test Size 0.2	0.916	0.917	0.911	0.914	0.982	0.982
Test Size 0.3	0.918	0.918	0.913	0.915	0.983	0.983
Test Size 0.4	0.918	0.919	0.912	0.916	0.982	0.983

Table 4. Gradient Boosting Model Performance Metrics



	Accuracy	Precision	Recall	F1-score	ROC-AUC	PR-AUC
Test Size 0.2	0.794	0.834	0.721	0.773	0.904	0.983
Test Size 0.3	0.788	0.83	0.712	0.767	0.9	0.983
Test Size 0.4	0.797	0.858	0.7	0.771	0.904	0.983

Table 5. CatBoost Model Performance Metrics



	Accuracy	Precision	Recall	F1-score	ROC-AUC	PR-AUC
Test Size 0.2	0.932	0.933	0.928	0.931	0.989	0.989
Test Size 0.3	0.933	0.935	0.926	0.931	0.989	0.989
Test Size 0.4	0.931	0.933	0.925	0.929	0.989	0.989

3. Model Curve Evaluations

In addition to performance metrics, the relationship between metrics and useful to analyze the success of models. Precision-recall curves display precision and recall at different thresholds. The ideal curve that hugs the top right corner of the graph, which displays few false positives and few false negatives. In this dataset, due to the even split, there is not a specific metric to optimize. A precision-recall curve that is close to diagonal indicates poor performance, which is close to random guessing. The Random Forest precision-recall curve in **Figure 7** is a strong performance across all test sizes.

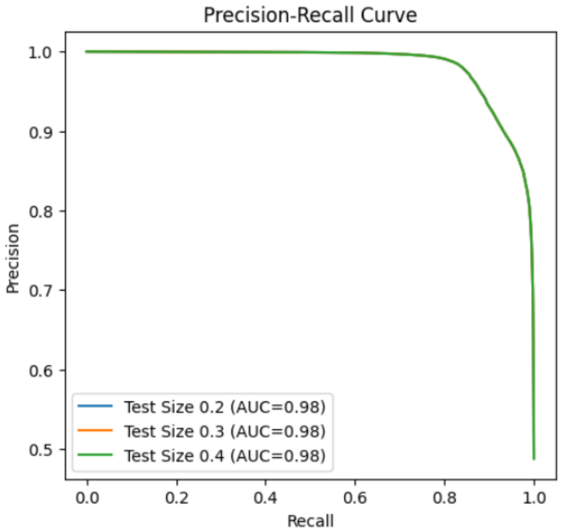


Figure 7. The Precision-Recall curve for Random Forest

While Random Forest shows high precision across a wide range of recall values, Gradient Boosting is not as strong. **Figure 8** has a jagged curve, with precision starting to drop at a recall of only 0.6. All of the test sizes overlap and display the same performance. Across all test sizes, Gradient Boosting struggles to maintain precision as it attempts to identify more positive cases. The model’s behavior is consistent but less stable than the Random Forest model.

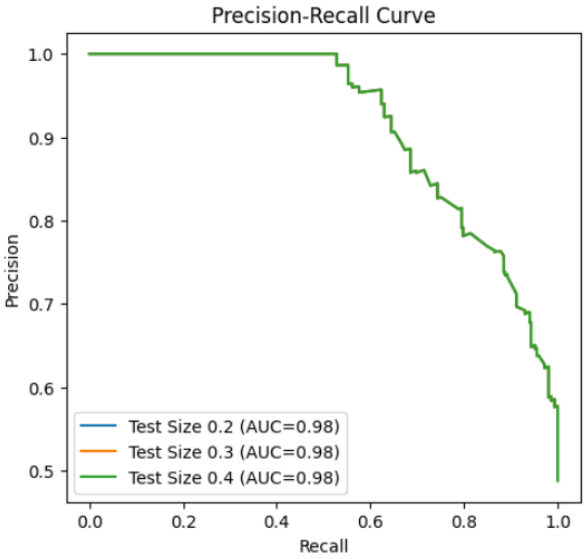


Figure 8. The Precision-Recall curve for Gradient Boosting

CatBoost in **Figure 9** displays an even higher precision than both other models for all test sizes, maintaining a strong performance until recall 0.8 where there is a slight decline. Although the curve is not as smooth as Random Forests', CatBoost still has higher precision. CatBoost is able to capture more true positives without sacrificing much precision. It is clearly the strongest in the precision-recall curve.

In addition to precision-recall curves, ROC curves are a great measure of performance. ROC curves measure true positive rate and false positive rate. A true positive rate should be high while the false positive rate should be low for a high performing model. On a curve, a good ROC will hug the top left corner. There is also a diagonal line that represents random guessing on an ROC curve. A good model should stay well above this line. Ideally, the curve should be a bent corner shape that rounds off. This is like a center- mirrored precision-recall curve.

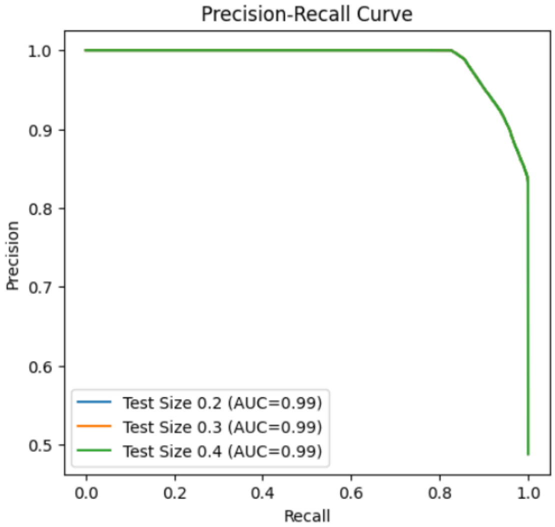


Figure 9. The Precision-Recall curve for CatBoost

The ROC curve for Random Forest in **Figure 10** shows very strong performance, with the curve starting to round off after a 0.8 true positive rate and returning at 0.95. It is consistent across all test sizes as well, indicating that the model has the ability to classify the two classes regardless of data size. The curve remained well above the diagonal line consistently, demonstrating a low false positive rate and high true positive rate across threshold. This suggests that Random Forest is highly effective at distinguishing between premium and non-premium products.

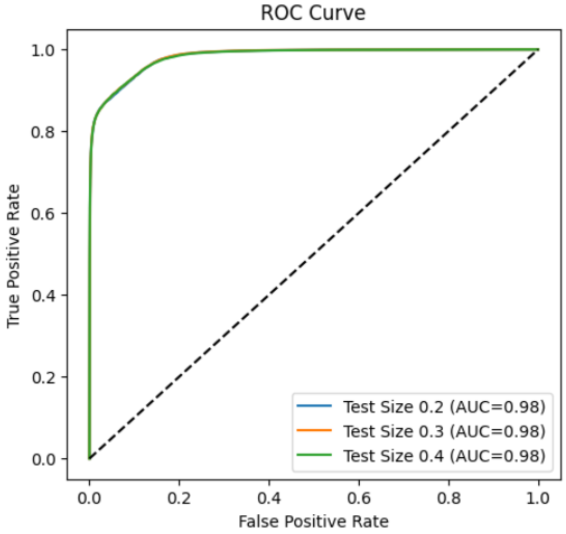


Figure 10. The ROC curve for Random Forest

The Gradient Boosting model did not perform well in the ROC curve, according to **Figure 11**. The false positive rate starts increasing when the true positive rate is just at 0.5. The curve is also very jagged throughout, suggesting instability and sensitivity to threshold changes. It is very easy to see the various test sizes as well, since the false positive rate does not appear to be consistent through the size difference. This gives a clear sign of model inconsistence due to varied training sizes. Overall, the shape of the curve shows that Gradient Boosting struggles to reliably classify premium vs non-premium products compared to all other models.

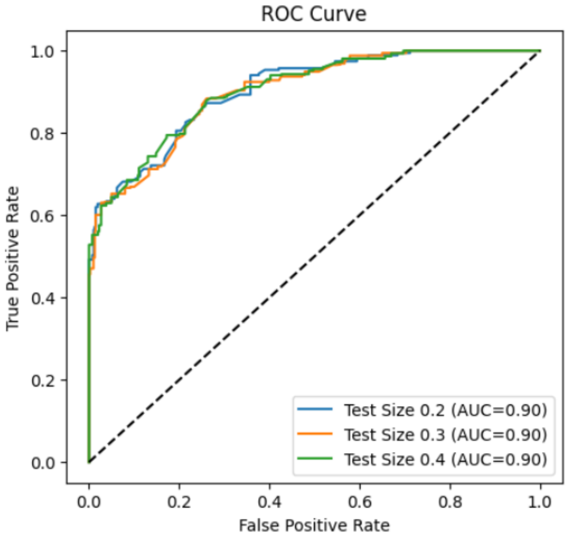


Figure 11. The ROC curve for Gradient Boosting

Unlike Gradient Boosting, CatBoost appears to have a strong ROC curve in **Figure 12**. This curve is very similar to the Random Forest curve, increasing the false positive rate after 0.85 true positive rate. It is again consistent across all test sizes, displaying model consistency. The curve is well above the diagonal line, with a low false positive rate and high true positive rate across threshold. Although the increasing false positive rate occurs slightly before Random Forests increases, the curve still suggests that CatBoost is highly effective at distinguishing between premium and non-premium products.

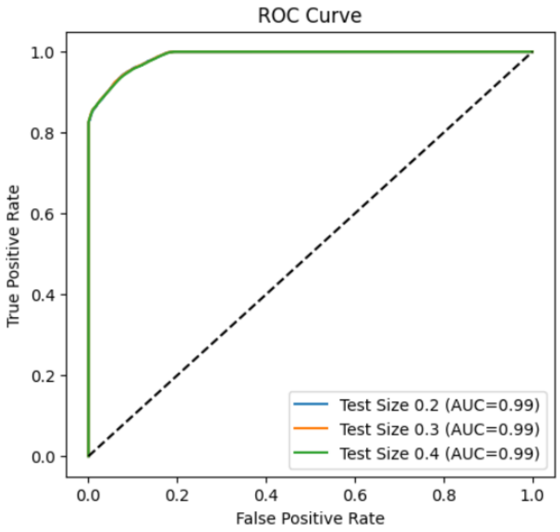


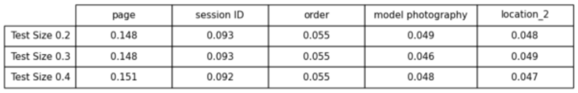
Figure 12. The ROC curve for CatBoost

4. Model Feature Importances

With all the metrics and curves established, feature importances can be used to explain the model's behavior. Feature importance scores for each model display which variables each model relies most on. It also displays which factors influence premium product clicks.

Random Forest ranks the most important features as page number, session ID, order, model photography, and location 2 (top middle of page) in **Table 6**. Page number and session ID are significantly higher ranked than the rest, meaning that the page number and the session were the biggest factors in predicting premium products. The value of pages that impact premium product clicks is the earlier pages. If a premium product is on page 1 or 2, customers are more likely to click on the product. As for session ID, it suggests that certain sessions are more likely to have premium products. This means customers either click on many premium products or very few, not an even mix.

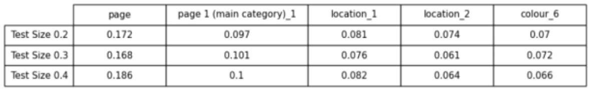
Table 6. Random Forest Model Feature Importance



	page	session ID	order	model photography	location_2
Test Size 0.2	0.148	0.093	0.055	0.049	0.048
Test Size 0.3	0.148	0.093	0.055	0.046	0.049
Test Size 0.4	0.151	0.092	0.055	0.048	0.047

Gradient Boosting has two of the same important features as Random Forest. Its top features include page number, product category, location 1 (top left), location 2 (top middle), and colour 6 (grey) from **Table 7**. Unlike Random Forest, all values are roughly similarly ranked for predicting premium products. The new features suggest that premium products with the category of trousers are more likely to be clicked. However, **Figure 3** displayed how category 1 of trousers is more likely to be clicked in general. Locations 1 and 2, top left and top middle, are also important. Yet, **Figure 5** displays how these locations are the most clicked locations already. Lastly, the colour grey is also important in this model. This may be useful, as the distribution of colors does not favor grey. Due to all of the factors which mimic the whole dataset distributions, the performance of the model is not particularly meaningful.

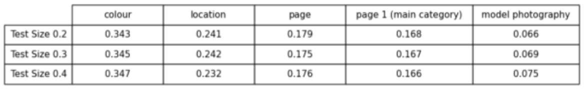
Table 7. Gradient Boosting Model Feature Importance



	page	page 1 (main category)_1	location_1	location_2	colour_6
Test Size 0.2	0.172	0.097	0.081	0.074	0.07
Test Size 0.3	0.168	0.101	0.076	0.061	0.072
Test Size 0.4	0.186	0.1	0.082	0.064	0.066

CatBoost has similar important features to Random Forest. The top five was colour, location on page, page number, product category, and model photography. For colour, olive, pink, red, and black are likely to be premium products. Olive was in fact premium 100% of the time. This displays that either most premium products are these colors, or those were the premium product colors that customers most clicked on. The product location on the page was the next factor. Clicked products in the top left and top right are more likely to be premium. This means either those are the main places that the site had their premium products, or that customers clicked on premium products that were visible in those locations. The page number suggested the same as Random Forest; earlier page numbers had more clicked on premium products. Main category was an important feature, just like Gradient Boosting. Lastly, model photography of en face (forwards facing models) were more likely to be clicked on premium products.

Table 8. CatBoost Model Feature Importance



	colour	location	page	page 1 (main category)	model photography
Test Size 0.2	0.343	0.241	0.179	0.168	0.066
Test Size 0.3	0.345	0.242	0.175	0.167	0.069
Test Size 0.4	0.347	0.232	0.176	0.166	0.075

All of these models display various important features which impact premium product prediction. CatBoost was by far the most accurate model, with the highest metrics and graph performance.

APPENDIX

CatBoost Classifier with a test size of 0.3 is the clear solution for predicting premium products with this dataset. Although all test sizes performed similarly, test size 0.3 was the best in nearly all metrics. It had an accuracy of 0.933, precision of 0.935, recall of 0.926, F1-score of 0.931, ROC-AUC of 0.989, and a PR-AUC of 0.989. Test size 0.3 had the best values for nearly all the metrics, while CatBoost had the best overall metrics compared to all other models. The test size of

0.3 is a good balance training set since it is large enough to train and sufficient for a good testing set. CatBoost was a strong model due to its categorical variable handling. It was easily able to use these values to predict and accurate premium products. Random Forest is still a strong model; however, it is less interpretable and harder with the number of categorical variables. Gradient Boosting was very much unstable and a more sensitive model. Although it was a decent model, its performance compared to both other models were subpar. CatBoost was able to handle categorical variables without one-hot encoding, and it remained very stable across many test sizes.

In applications, this model helps businesses determine which product clicks are premium and understand which product features strongly drive customers to click on premium products for better targeting and product strategies. By analyzing a successful model's feature importance, businesses can see which attributes directly influence customer behavior. Companies are able to optimize marking strategies, adjust product placement, and spend less on premium advertisements. Moreover, understanding the features helps with anticipating customer preferences, improving user experiences, and increasing click rates on premium products.

REFERENCES

[1] “Clickstream Data for Online Shopping.” 2019. *UCI Machine Learning Repository*. DOI: https://doi.org/10.24432/C5QK7X.

[2] IBM. 2021. What Is Random Forest? IBM. https://www.ibm.com/think/topics/random-forest

[3] B. Clark and F. Lee. 2025. What Is Gradient Boosting? IBM. https://www.ibm.com/think/topics/gradient-boosting

[4] O. Jeremiah. 2024. CatBoost in Machine Learning: A Detailed Guide. DataCamp. https://www.datacamp.com/tutorial/catboost

GENERATIVE AI PROMPTS USED

1. What are the top 5 ML classification models most commonly used and why?
2. What are the key factors from this rubric [insert project evaluation] that I should focus heavily on in my report?
3. What does this bug [insert python bug] mean?